tidyverse: pipe variations

Donald Butler

11/07/2021

Tidyverse pipe operator

Comparing Student Performance on Exams

Source: https://www.kaggle.com/spscientist/students-performance-in-exams

I selected a simple dataset that could be used to demonstrate the use of the pipe operator to non-tidyverse functions.

```
student_performance = read_csv("https://raw.githubusercontent.com/dab31415/DATA607/main/StudentsPerforma
```

```
## Rows: 1000 Columns: 8
```

```
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr (5): gender, race/ethnicity, parental level of education, lunch, test pr...
## dbl (3): math score, reading score, writing score
```

```
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```
student_performance
```

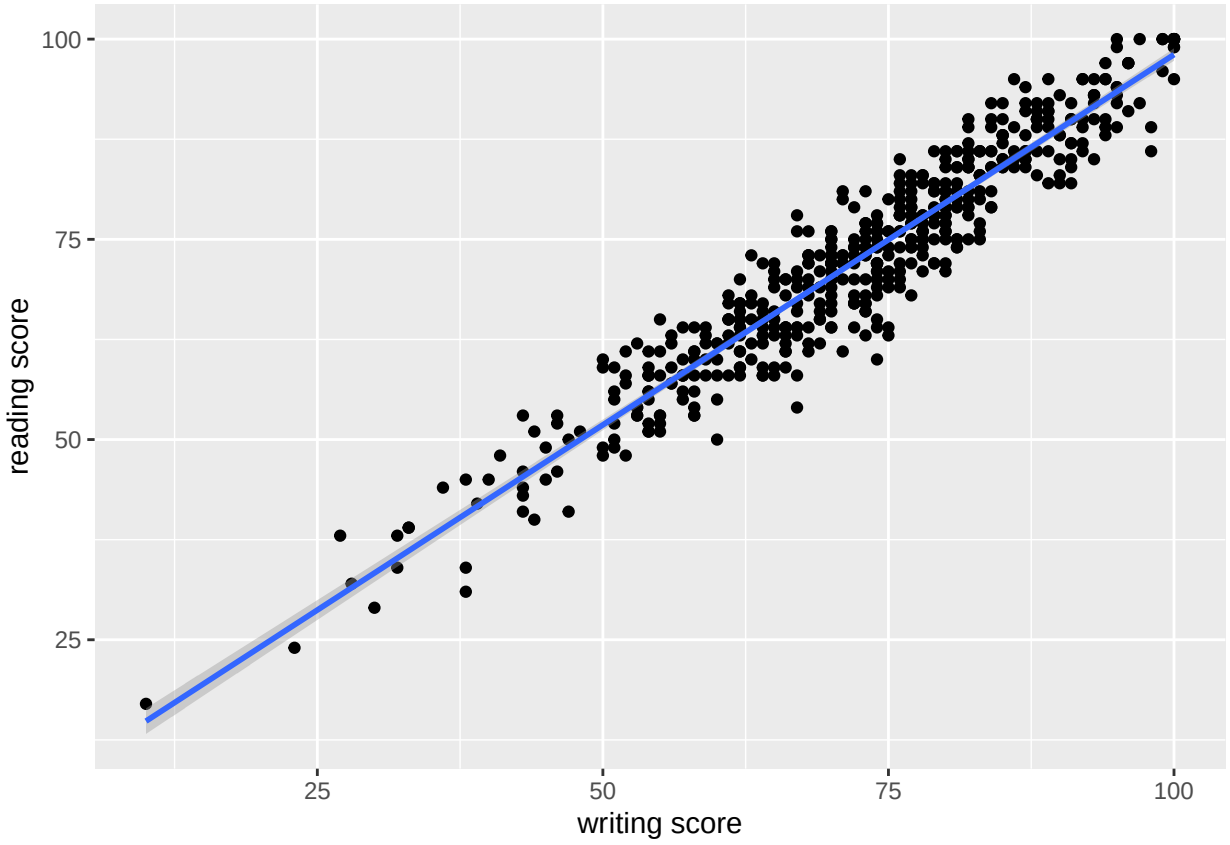
```
## # A tibble: 1,000 x 8
##    gender `race/ethnicity` `parental level ~ lunch `test preparati~ `math score`
##    <chr>  <chr>            <chr>             <chr> <chr>                    <dbl>
##  1 female group B          bachelor's degree stan~ none                        72
##  2 female group C          some college      stan~ completed                   69
##  3 female group B          master's degree   stan~ none                        90
##  4 male   group A          associate's degr~ free~ none                        47
##  5 male   group C          some college      stan~ none                        76
##  6 female group B          associate's degr~ stan~ none                        71
##  7 female group B          some college      stan~ completed                   88
##  8 male   group B          some college      free~ none                        40
##  9 male   group D          high school       free~ completed                   64
## 10 female group B          high school       free~ none                        38
## # ... with 990 more rows, and 2 more variables: reading score <dbl>,
## #   writing score <dbl>
```

1A: basic pipe

The magrittr package defines the pipe operator, %>%, which is heavily used in the tidyverse. The pipe operator allows you to pass the output of one function as the input of the next. This works well with tidyverse functions because the data parameter is the first one in the tidyverse functions.

```
student_performance %>%
  filter(gender == 'female') %>%
  ggplot(aes(`writing score`,`reading score`)) +
  geom_point() +
  geom_smooth(method = lm)
```

```
## `geom_smooth()` using formula 'y ~ x'
```



1B: How can I pipe to a function where data is not the first parameter?

The lm function is used to perform linear regression. The first parameter of lm is the formula to perform the regression on, and the second parameter is data. The magrittr package provides for an argument placeholder, ., to allow the pipe to pass to any parameter in the next function.

```
student_performance %>%
  filter(gender == 'female') %>%
  lm(`reading score` ~ `writing score`, data = .)
```

```
##
## Call:
## lm(formula = `reading score` ~ `writing score`, data = .)
##
## Coefficients:
##     (Intercept)  `writing score`
##          5.5965           0.9247
```